A Multi-view CNN-based Acoustic Classification System for Automatic Animal Species Identification

Weitao Xu[a], Xiang Zhang[b], Lina Yao[b], Wanli Xue[b], Bo Wei[c]

[a]*Department of Computer Science, City University of Hong Kong, Hong Kong*
[b]*School of Computer Science and Engineering, University of New South Wales, Australia*
[c]*Department of Computer and Information Sciences, Northumbria University, UK*

Abstract

Automatic identification of animal species by their vocalization is an important and challenging task. Although many kinds of audio monitoring system have been proposed in the literature, they suffer from several disadvantages such as non-trivial feature selection, accuracy degradation because of environmental noise or intensive local computation. In this paper, we propose a deep learning based acoustic classification framework for Wireless Acoustic Sensor Network (WASN). The proposed framework is based on cloud architecture which relaxes the computational burden on the wireless sensor node. To improve the recognition accuracy, we design a multi-view Convolution Neural Network (CNN) to extract the short-, middle-, and long-term dependencies in parallel. The evaluation on two real datasets shows that the proposed architecture can achieve high accuracy and outperforms traditional classification systems significantly when the environmental noise dominate the audio signal (low SNR). Moreover, we implement and deploy the proposed system on a testbed and analyse the system performance in real-world environments. Both simulation and real-world evaluation demonstrate the accuracy and robustness of the proposed acoustic classification system in distinguishing species of animals.

Keywords: Wireless acoustic sensor network, Animal identification, Deep learning, CNN

1. Introduction

Wireless Acoustic Sensor Network (WASN) based animal monitoring is of great importance for biologists to monitor real-time wildlife behavior for long periods and under variable weather/climate conditions. The acquired animal voice can provide valuable information for researchers, such as the density and diversity of different species of animals [1, 2, 3]. For example, Hu et al. proposed a WASN application to census the populations of native frogs and the invasive introduced species (Cane Toad) in Australia [4]. There are also several important commercial applications of acoustic animal detection. For instance, America imports billions of dollars of timber from Aisa every year. However, the inadvertent introduction of the Asian Longhorn Beetle has cost USA government millions of dollars to eradicate the Beetle population [5]. Therefore, a wireless monitoring system is imperative to detect the distribution of these insects.

There are a large volume of audio monitoring systems in the literature [4, 6, 7, 8, 9, 10, 11, 12, 13, 14]. In the early stage, biologists have traditionally deployed audio recording systems over the natural environment where their research projects were developed [6, 7]. However this procedure requires human presence in the area of interest at certain moments. In recent years, with the development of WSN, some researchers have proposed remotely accessible systems in order to minimize the impact of the presence of human beings in the habitat of interest [4, 12, 13, 14].

Despite much effort in this area, previous studies suffer from several disadvantages. First, traditional methods usually first extract a number of appropriate features and then employ classic machine

Email addresses: weitaoxu@cityu.edu.hk (Weitao Xu), xiang.zhang3@student.unsw.edu.au (Xiang Zhang), lina.yao@unsw.edu.au (Lina Yao), w.xue@unsw.edu.au (Wanli Xue), bo.wei@northumbria.ac.uk (Bo Wei)

learning methods such as Support Vector Machine (SVM) or K-Nearest Neighbours (KNN) to detect the species of the animals. Features, such as statistical features through statistical anlaysis (e.g., variance, mean, median), Fast Fourier Transmission (FFT) spectrum, spectrograms, Wigner-Ville distribution (WVD), Mel-frequecy cepstrum coefficient (MFCC) and wavelets have been broadly used. However, extracting robust features to recognizing noisy field recordings is non-trivial. While these features may work well for one , it is not clear whether they generalize to other species. The specific features for one application do not necessarily generalize to others. Moreover, a significant number of calibrations are required for both manually feature extraction and the classification algorithms. This is because the performance of the traditional classifiers such as SVM and KNN [8, 10, 11] highly depends on the quality of the extracted features. However, handcrafting features relies on a significant amount of domain and engineering knowledge to translate insights into algorithmic methods. Additionally, manual selection of good features is slow and costly in effort. Therefore, these approaches lack scalability for new applications. Deep learning technologies can solve these problems by using deep architectures to learn feature hierarchies. The features that are higher up in deep hierarchies are formed by the composition of features on lower levels. These multi-level representations allow a deep architecture to learn the complex functions that map the input (such as digital audio) to output (e.g. classes), without the need of dependence on manual handcrafted features.

Secondly, these approaches suffer from accuracy degradation in real-world applications because of the impact of environmental noise. The voice recorded from field usually contains much noise which poses a big challenge to real deployment of such system. To address this problem, Wei et al. [15] proposed an *in-situ* animal classification system by applying sparse representation-based classification (SRC). SRC uses ℓ_1-optimization to make animal voice recognition robust to environmental noise. However, it is known that ℓ_1-optimization is computationally expensive [16, 17], which limits the application of their system in resource-limited sensor nodes. Additionally, in order to make SRC achieve high accuracy, a large amount of training data is required. This means a wireless sensor node can only store a limited number of training classes because of the limited storage.

Recently, deep learning has emerged as a powerful tool to solve various recognition tasks such as face recognition [18], human speech recognition [19, 20] and natural language processing [21]. The application of deep learning in audio signal is not new; however, most previous studies focus on human speech analysis to obtain context information [19, 20, 22]. Limited efforts have been devoted to applying deep learning in WASN to classify different species of animals. To bridge this gap, we aim to design and implement a acoustic classification framework for WASN by employing deep learning techniques. Convolutional Neural Network (CNN), as a typical deep learning algorithm, has been widely used in high-level representative feature learning. In detail, CNN is enabled to capture the local spatial coherence from the input data. In our case, the spatial information refers to the spectral amplitude of the audio signal. However, one drawback of the standard CNN structure is that the filter length of the convolution operation is fixed. As a result, the convolutional filter can only discover the spatial features with the fixed filter range. For example, CNN may explore the short-term feature but fail to capture the middle- and long-term features. In this paper, we propose a multi-view CNN framework which contains three convolution operation with three different filter length in parallel in order to extract the short-, middle-, and long-term information at the same time. We conduct extensive experiments to evaluate the system on real datasets. More importantly, we implement the proposed framework on a testbed and conduct a case study to analyse the system performance in real environment. To the best of our knowledge, this is the first work that designs and implements a deep learning based acoustic classification system for WASN.

The main contributions of this paper are threefold:

- We design a deep learning-based acoustic classification framework for WASN, which adopts a multi-view convolution neural network in order to automatically learn the latent and high-level features from short-, middle- and long-term audio signals in parallel.
- We conduct extensive evaluation on two real dataset (Forg dataset and Cricket dataset) to demonstrate the classification accuracy and robustness of the proposed framework to environmental noise. Evaluation results show that the

proposed system can achieve high recognition accuracy and outperform traditional methods significantly especially in low SNR scenarios.

- We implement the proposed system on a testbed and conduct a case study to evaluate the performance in real world environments. The case study demonstrate that the proposed framework can achieve high accuracy in real applications.

The rest of this paper is organized as follows. Section 2 introduces related work. Then, we describe system architecture in Section 3 and evaluate the system performance in Section 4. We implement the system on a testbed and conduct user study to evaluate the system in Section 5. Finally, Section 6 concludes the paper.

2. Related Work

Animal voice classification has been extensively studied in the literature. At the highest level, most work extract sets of features from the data, and use these features as inputs for standard classification algorithms such as SVM, KNN, decision tree, or Bayesian classifier. Previous studies have involved a wide range of species which include farm animals [23], bats [24], birds [8, 11, 25], pests [26], insects [27] and anurans [28]. The works of Anderson et al. [6] and Kogan et al. [7] were among the first attempts to recognize bird species automatically by their sounds. They applied dynamic time warping and hidden Markov models for automatic song recognition of Zebra Finche and Indigo Punting. In [2], the authors focus on classifying two anuran species: Alytes obstetricans and Epidalea calamita using generic descriptors based on an MPEG-7 standard. Their evaluation demonstrate that MPEG-7 descriptors are suitable to be used in the recognition of different patterns, allowing a high scalability. In [1], the authors propose to classify animal sounds in the visual space, by treating the texture of animal sonograms as an acoustic fingerprint. Their method can obviate the complex feature selection process. They also show that by searching for the most representative acoustic fingerprint, they can significantly outperform other techniques in terms of speed and accuracy.

The WSN has been massively applied in sensing the environment and transferring collected samples to the server. However, it is challenging to realize in-network classification system because of the limited computational ability of wireless sensor node. Recently, several research works regarding in-network classification have been proposed. Sun et al. [29] dynamically select the feature space in order to accelerate the classification process. A hybrid sensor networks is designed by Hu et al. [4] for in-network and energy-efficient classification in order to monitor amphibian population. Wei et al. [15] proposed a sparse representation classification method for acoustic classification on WSN. A dictionary reduction method was designed in order to improve the classification efficiency. The sparse representation classification method was also used by face recognition on resource-constrained smart phones to improve the classification performance [16, 17].

Deep learning has achieved great success over the past several years for the excellent ability on high-level feature learning and representative information discovering. Specifically, deep learning has been widely used in a number of areas, such as computer version [30], activity recognition [31, 32], sensory signal classification [33, 34, 35], and brain computer interface [36]. Wen et al. [30] propose a new supervision signal, called center loss, for face recognition task. The proposed center loss function is demonstrated to enhance the discriminative power of the deeply learned features. Chen et al. [31] propose an interpretable parallel recurrent neural network with convolutional attentions to improve the activity recognition performance based on Inertial Measurement Unit signals. Zhang et al. [33] combine deep learning and reinforcement learning to deal with multi-modal sensory data (e.g., RFID, acceleration) and extract the latent information for better classification. Recently, deep learning involves in the brain signal mining in brain computer interface (BCI). Zhang et al. [36] propose an attention-based Encoder-Decoder RNNs (Recurrent Neural Networks) structure in order to improve the robustness and adaptability of the brainwave based identification system.

There are also several works that apply deep learning techniques in embedded devices. Lane et al. [37] propose low-power Deep Neural Network (DNN) model for mobile sensing. CPU and DSP in one mobile device are exploited for activity recognition. Lane et al. [37] also design a DNN model for audio sensing in mobile phone by using dataset from 168 places for the training purpose. A framework DeepX is further proposed for software accelerating

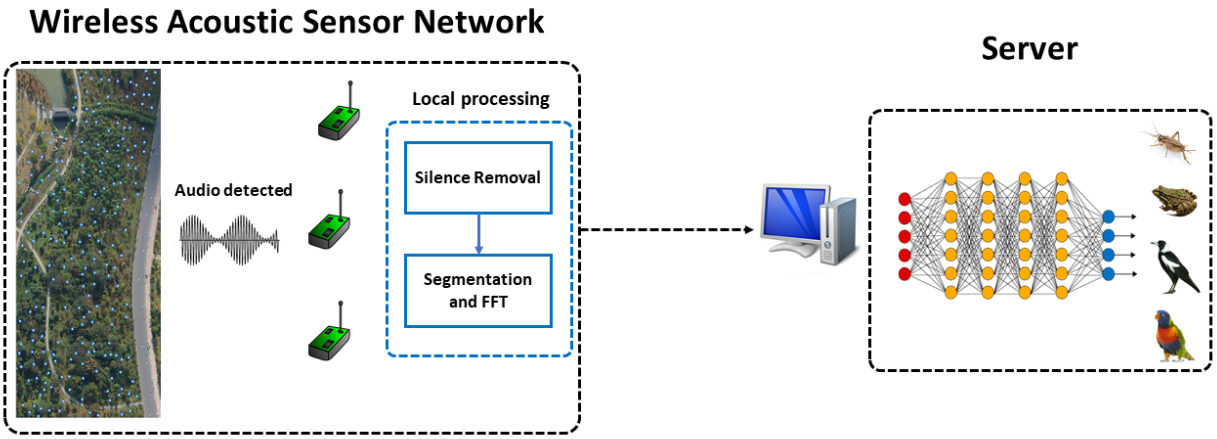


Figure 1: System Overview.

on mobile devices [38].

In terms of animal voice classification, Zhang et al. [39], Oikarinen et al. [40] study animal voice classification using deep learning techniques. Our method is different from these two works. Our studies focus on voice classification in noisy environment while the voice data in [39] are collected from controlled room without environmental noise. Instead of classifying different animals, [40] analyses different call types of marmoset monkeys such as Trill, Twitter, Phee and Chatter. Moreover, we implement the proposed system on a testbed and evaluate its performance in real world environment. In another work [41], Stavros Ntalampiras used transfer learning to improve the accuracy of bird classification by exploiting music genres. Their result show that the transfer learning scheme can improve classification accuracy by 11.2%. Although the goal of this work and our study is to improve the recognition accuracy with deep learning technology, the methodologies are different. Our approach analyses the inherent features of audio signal and propose a multi-view CNN model to improve the accuracy. Instead of looking at the bird audio signals alone, Stavros Ntalampiras proposed to statistically analyse audio signals using their similarities with music genres. Their method, however, is only effective for a limited number of bird species because they need to perform feature transformation again when a new bird species comes in. In comparison, our approach is applicable for a large number of bird species. A number of studies also apply deep learning technologies in bird voice classification [42, 43, 44], however, they only use conventional deep learning approaches such as CNN and do not make any novel improvement. In this paper, we propose a multi-view CNN model and evaluation results show that the proposed model outperforms the conventional CNN.

3. System Design

3.1. System Overview

As shown in Figure 1, our proposed framework consists of two parts: WASN and server. In the WASN, the wireless nodes will detect and record animal voices and then perform local processing which include silence removal, segmentation and FFT. We process signal *in-situ* before uploading because of the high sampling frequency of audio signal and energy inefficiency of wireless communication [45, 15]. The spectrum signal obtained from FFT can save half spaces since FFT is symmetric. On the server side, the spectrum signal will be fed into a deep neural network to obtain the species of the animal. The classification results can be used by biologists to analyze the density, distribution and behavior of animals.

Wireless sensors are usually resource-poor relative to server, and not able to run computationally expensive algorithms such as deep learning models. Therefore, we assume all the wireless sensors can connect to a server via wireless communication technologies, such as ZigBee, Wi-Fi, and LoRa [46]. However, there may be network failure, server failure, power failure, or other disruption makes offload

impossible. While such failures will hopefully be rare, they cannot be ignored in a cloud-based system. In this case, the node can transmit the data to the gateway or a nearby server which are usually resource-rich and capable of running deep learning models. Alternatively, the classification can be performed in the node to recognize only a few species, pre-defined by the user. When offloading becomes possible again, the system can revert to recognizing its full range of species.

In the following parts, we will describe the design details of each component.

3.2. Local Processing

Silence Removal. The collected audio signal usually contains a large amount of silent signal when this is no animal present. Therefore, we apply a simple silence removal method on the raw signal to delete not-of-interest area. The procedure is explained in Algorithm 1. We first calculate the root mean square (RMS) of each window which contains 1s samples and then compare it with a pre-defined threshold learned from the environment. The windows of samples whose RMS above the threshold will be kept. The threshold is determined by exhaustive search. To be specific, we increase the threshold from 0 to 0.5 with an increment of 0.01, then choose the one that can achieve the best performance (0.03 in this paper).

Algorithm 1 Silence Removal

1: **Input**: Audio Segment $S_{i=1:N} \in \mathbb{R} > 1$, where N is the total number of segments and ρ is the threshold
2: **for** $i = 1 : N$ **do**
3: **if** RMS $(S_i) < \rho$ **then**
4: Remove (S_i)
5: **end if**
6: **end for**

Figure 2 shows an example of silence removal on an animal voice recording. We can see that it can effectively remove the silent periods and detect the present of animals.

Segmentation and FFT. After silence removal, we obtain audio signals containing animal vocalization only. The audio signal is segmented into consecutive sliding windows with 50% overlap. Hamming window is used in this paper to avoid spectral leakage. Each window contains 2^{14} samples which

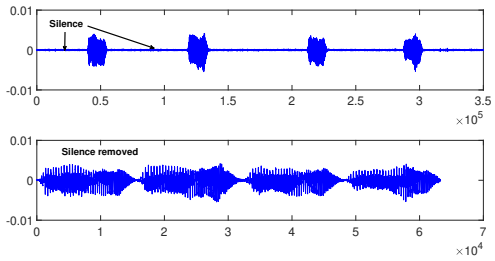


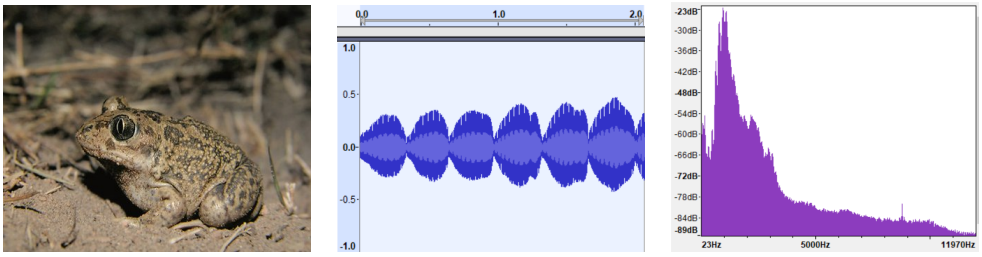
Figure 2: Silence removal.

is chosen to balance the trade-off between classification accuracy and latency as discussed in Section 4. The overlap in sliding window is used to capture changes or transitions around the window limits. Then we perform FFT on each segment to calculate spectrum energy (i.e. the magnitude of the FFT coefficients). As an example, Figure 3 shows the sound in time and frequency domain of two different frog species: Cultripes and Litoria Caerulea. It is conspicuous that they have different spectrum distributions. The graphs are plotted by audio signal analysis software Audacity.

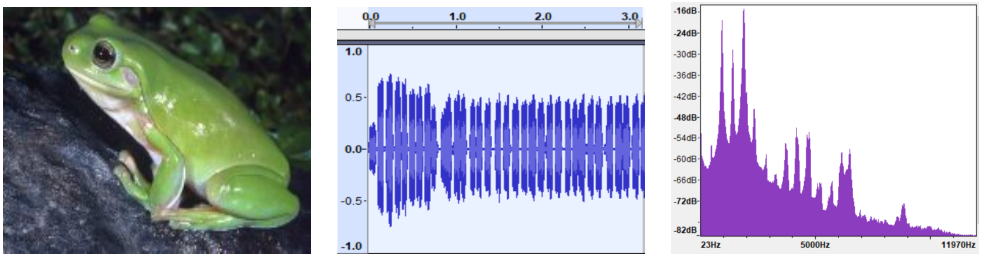
3.3. Multi-view Convolutional Neural Networks

We propose a deep learning framework in order to automatically learn the latent and high-level features from the processed audio signals for better classification performance. Among deep learning algorithms, CNN is widely used to discover the latent spatial information in applications such as image recognition [47], ubiquitous [48], and object searching [49], due to their salient features such as regularized structure, good spatial locality and translation invariance. CNN applies a convolution operation to the input, passing the result to the next layer. Specifically, CNN captures the distinctive dependencies among the patterns associated to different audio categories. However, one drawback of the standard CNN structure is that the filter length of the convolution operation is fixed. As a result, the convolutional filter can only discover the spatial features with the fixed filter range. For example, CNN may explore the short-term feature but fail to capture the middle- and long-term features.

To address the mentioned challenge, we propose a multi-view CNN framework which applies three different filter length to extract the short-, middle-, and long-term features in parallel. As shown in



(a) Cultripes (b) Cultripes: time domain (c) Cultripes: spectrum



(d) Litoria Caerulea (e) Litoria Caerulea: time domain (f) Litoria Caerulea: spectrum

Figure 3: Audio signal of two species of frog.

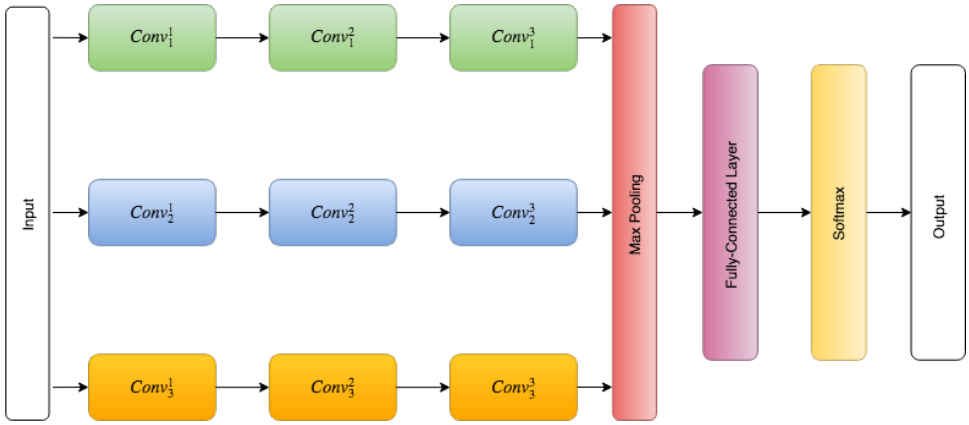


Figure 4: Multi-view CNN workflow.The audio signal is feed into three views for short-term, middle-term, and long-term spatial dependencies learning. $Conv_h^k$ denotes the k-th convolutional operation in the h-th view. The learned features from the multi-view structure are processed by the max pooling layer for dimension reduction, which are followed by the fully-connected layer, softmax layer, and at last predict the animal species as output.

Figure 4, the proposed framework regards the processed audio signals as input and feed into three views at the same time. Each view contains three convolutional layers. $Conv_h^k$ denotes the k-th convolutional operation in the h-th view. The convolutional layer contains a set of filters to convolve the audio data followed by the nonlinear transformation to extract the geographical features. The filter length keeps invariant in the same view while varies in different views. The extracted features from the multi-view pipe are stacked together and then through the max pooling operation for dimension reduction. Afterward, a fully-connected layer, a softmax layer and the output layer work as a classifier to predict the audio label. The proposed multi-view CNN has several key differences from the inception module [50] although the ideas are similar. First, [50] has a 1×1 convolutional filter in the module in order to prevent the information corruption brought by inter-channel convolutions.

The proposed multi-view CNN does not has this component. This is because in our case the input data are naturally formed as a vector which represents the spectral information of the acoustic signals. Moreover, [50] adds an alternative parallel pooling path in the middle layer to acquire additional beneficial effect. However, we believe this may cause information loss and only perform the pooling operation after the concentration of the results of various views.

Suppose the input audio data $\boldsymbol{E}$ has shape $[M, L]$ with depth as 1. The chosen three convolutional filters with size in short-, middle-, and long-term views are $[M, 10]$, $[M, 15]$, $[M, 20]$, respectively. The stride sizes keep $[1, 1]$ for all the convolutional layers. The stride denotes the x-movements and y-movements distance of the filters. Since the audio signals are arranged as 1-dimension data, we set $M = 1$. Same shape zero padding is used, which keeps the sample shape constant during the the convolution calculation. In the convolutional operation, the feature maps from the input layer are convolved with the learnable filters and fed to the activation function to generate the output feature map. For a specific convolutional area (also called perceptive area) $\boldsymbol{x}$ which has the same shape as the filter, the convolutional operation can be described as

$$\boldsymbol{x}' = tanh(\sum_i \sum_j \boldsymbol{f}_{ij} * \boldsymbol{x}_{ij})$$

where $\boldsymbol{x}'$ denotes the filtered results while $\boldsymbol{f}_{ij}$ denotes the i-th row and the j-th column element in the trainable filter. We adopt the widely used *tanh* activation function for nonlinearity. The depth of input sample transfers to D through the convolutional layer and the sample shape is changed to $[M, L, D]$. In particular, the corresponding depth $D_h = 2, 4, 8$ for three convolutional layers. The features learned from the filters are concatenated and flattened to $[1, M * L * \sum_{h=1}^{3} D_h]$. The max pooling has $[1, 3]$ as both pooling length and strides. Therefore, the features with shape $[1, M * L * \sum_{h=1}^{3} D_h/3]$ after the pooling operation, which are forwarded to the fully-connected layer. The operation between the fully-connected layer and the output layer can be represented by

$$\boldsymbol{y} = softmax(\bar{\boldsymbol{w}}\boldsymbol{E}^{\boldsymbol{FC}} + \bar{\boldsymbol{b}})$$

where FC denotes the fully-connected layer while the $\bar{\boldsymbol{w}}$ and $\bar{\boldsymbol{b}}$ denote the corresponding weights matrix and biases. The softmax function is used for activation. For each sample, the corresponding label information is presented by one-hot label $\boldsymbol{y} \in \mathbb{R}^H$ where H denotes the category number of acoustic signals. The error between the predicted results and the ground truth is evaluated by cross-entropy

$$loss = -\sum_{h=1}^{H} \boldsymbol{y}_h log(p_h)$$

where p_h denotes the predicted probability of observation of an object belonging to category h. The calculated error is optimized by the AdamOptimizer algorithm [51]. To minimize the possibility of overfitting, we adopt the dropout strategy and set the drop rate to 80%.

4. Evaluation

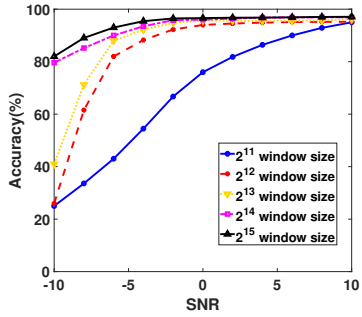
4.1. Goals, Metrics, and Methodology

In this section, we evaluate the performance of the proposed system based on two real datasets. The goals of the evaluation are twofold: 1) evaluate the performance of the proposed system under different settings; 2) compare the proposed system with previous animal vocalization system. We use two datasets collected from real-world for evaluation. The first dataset contains audio signals recorded from fourteen different species of frogs. The sampling frequency for this dataset is 24Khz. More details about this dataset can be found in [15]. The second dataset [1] contains audio signals recorded from different species of crickets. The data consists of twenty species of crickets, eight of which are Gryllidae and twelve of which are Tettigoniidae. The sampling frequency is also 24Khz. More details about this dataset can be found in [1]. For completeness, Table. 1 lists all the species we used in the experiments. In this paper, we use SVM and KNN to benchmark ASN classification because they have been widely used in WASN classification systems [8, 10, 11]. We evaluate the performance of SVM and KNN by using frequency domain and Mel-frequency cepstral coefficients (MFCCs), respectively. The parameters in SVM and KNN are well tuned to give highest accuracy. In addition, we compare the accuracy of our system with a recent work which is based on SRC [15] and conventional CNN. In total, we compare our method

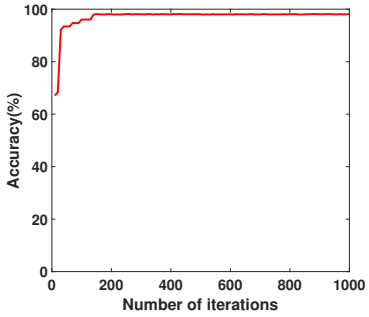
[1] http://alumni.cs.ucr.edu/ yhao/animalsoundfingerprint.html

Table 1: Species used in the experiments.

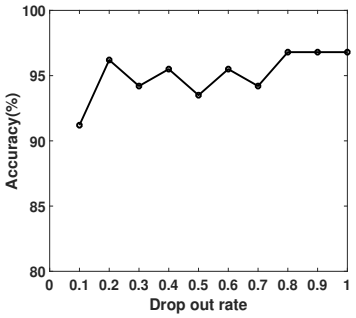
Frog dataset		Cricket dataset (1 belongs to Gryllidae, 2 belongs to Tettigoniidae)	
Cyclorana Cryptotis	Cyclorana Cultripes	Acheta[1]	Aglaothorax[2]
Limnodynastes Convexiusculus	Litoria Caerulea	Allonemobius[1]	Amblycorypha[2]
Litoria Inermis	Litoria Nasuta	Anaxipha[1]	Anaulacomera[2]
Litoria Pallida	Litoria Rubella	Anurogryllus[1]	Arethaea[2]
Litoria Tornieri	Notaden Melanoscaphus	Cyrtoxipha[1]	Atlanticus[2]
Ranidella Bilingua	Ranidella Deserticola	Eunemobius[1]	Belocephalus[2]
Uperoleia Lithomoda	Bufo Marinus	Gryllus[1]	Borinquenula[2]
		Hapithus[1]	Bucrates[2]
		Capnobotes[2]	Caribophyllum[2]
		Ceraia[2]	Conocephalus[2]



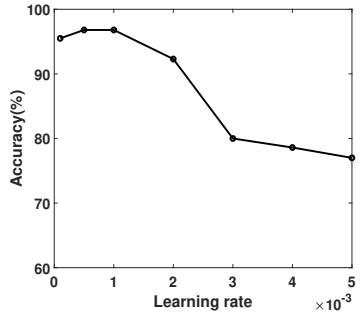
(a) Impact of window size.



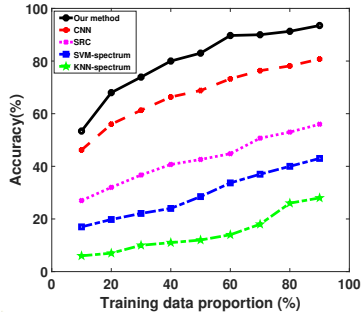
(b) Impact of iterations.



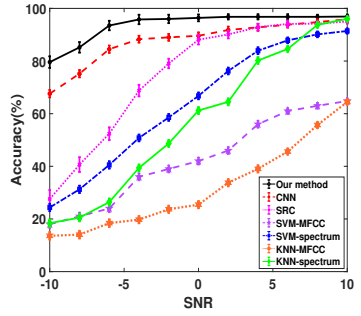
(c) Impact of dropout rate.



(d) Impact of learning rate.



(e) Impact of training dataset.



(f) Comparison with other methods.

Figure 5: Evaluation results of frog dataset.

with six classifiers: CNN, SRC, SVM-MFCC, SVM-spectrum, KNN-MFCC and KNN-spectrum. For each classifier, we perform 10-fold cross-validation on the collected dataset. In the original dataset, the data only contain little environment noise. Therefore, to demonstrate the robustness of the proposed framework, we add different scales of environmental noise to create different SNRs. This is used to simulate the real environment because the recorded animal voices are usually deteriorated by environmental noise in real WASN. In the evaluation, it is done by adding different scales of random Gaussian noise to the original audio data. In this paper, we focus on the following four metrics: *accuracy*, *precision*, *recall* and *F1-score*. We plot the results of the average values and stand deviation obtained from 10 folds cross-validation.

4.2. Performance of Frog Dataset

4.2.1. Impact of parameters

We first evaluate the impact of important parameters in our system. On the node's side, the important parameters include window size of segment. On the server's side, the important parame-

Table 2: Performance of different methods on frog dataset (SNR=-6dB).

	Our method	CNN	SRC	SVM-MFCC	SVM-Spectrum	KNN-MFCC	KNN-Spectrum
Accuracy	**94.7%**	82.7%	53.4%	24.4%	40.5%	20.1%	26.4%
Precision	**93.1%**	81.6%	54.2%	25.9%	43.5%	19.9%	25.1%
Recall	**94.3%**	82.4%	53.7%	24.7%	41.2%	21.5%	27.1%
F1-score	**92.9%**	81.2	52.1%	25.1%	39.6%	20.7%	25.7%

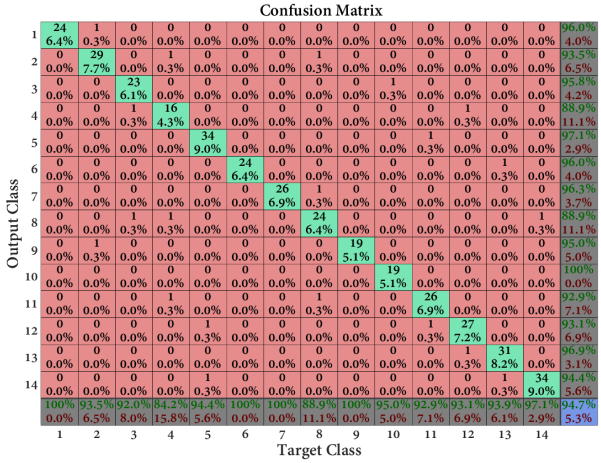


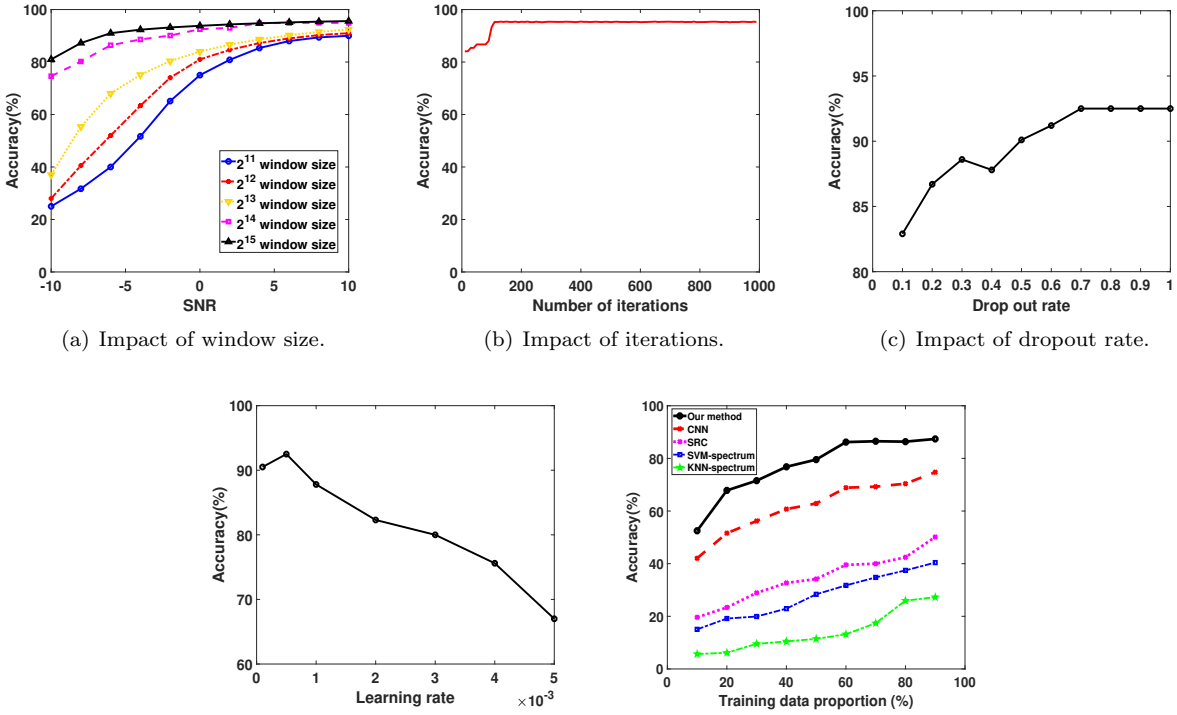
Figure 6: Confusion matrix of frog dataset.

ters include the number of iterations in training, the dropout rate and learning rate in CNN, the size of training dataset. Dropout is a technique where randomly selected neurons are ignored during training. For example, the dropout rate of 80% means that we randomly select 20% of the neurons and drop them (force the values of them as 0). The dropout strategy is widely used to enhance the generalization of a machine learning model and prevent overfitting. The learning rate is a hyper-parameter that controls how much we are adjusting the weights of the neuron network with respect to the loss gradient.

To evaluate the impact of window size, we vary the window size from 2^{11} to 2^{15} samples and calculate the accuracy of our scheme. From the results in Figure 5(a), we can see that there is a performance gain when we increase the window size and the improvement reduces after 2^{14} samples. Although we can achieve higher accuracy with more samples, the resource consumption of FFT operation which runs on the wireless sensor node also increases. Therefore, we choose to use 2^{14} window size to balance the trade-off between accuracy and resource consumption.

Figure 5(b) shows the accuracy along with different training iterations. We can see that the proposed method converges to its highest accuracy in less than 200 iterations. The results show that the proposed framework can finish training quickly. Figure 5(c) plots the accuracy of various dropout rates. We can observe that the accuracy fluctuates first and then becomes stable after the dropout rate is greater than 0.8. Therefore, we set the default dropout rate to be 0.8. Moreover, we can infer from Figure 5(c) that our model is not very sensitive to the dropout rate. This is because the Frog dataset matches well with the proposed multi-view CNN, as a result, the convergence suffers less from overfitting which can be demonstrated by the good convergence property as shown in Figure 5(b). Figure 5(d) shows the accuracy under different learning rates. We can see that it achieves the highest accuracy when the learning rate is 0.5×10^{-3} and 1×10^{-3}. Correspondingly, we choose 0.001 to reduce the training time because the smaller the learning rate is, the slower the training process is. From Figure 5(d), we can observe that the performance varies dramatically with the increasing of learning rate. One possible reason for this is that the gradient surface of our loss function is not smooth and very sensitive to the learning rate. The optimiser is easy to step over the local optima while the learning rate is larger than a threshold.

Next, we evaluate the accuracy of the proposed system under different sizes of training dataset. In this experiment, we use different proportions of the whole dataset for training, and use the left dataset for testing. The proportion increases from 10% to 90% with an increment of 10%. For example, the proportion of 10% means we use 10% of the dataset for training, and use the left dataset for testing. For comparison purpose, we also calculate the accuracy of CNN, SRC, SVM and KNN. From the results in Figure 5(e), we can see that our method continuously achieves the highest accuracy, and the accuracy becomes relatively stable after 60% of the dataset is used for training. We also notice that



(a) Impact of window size. (b) Impact of iterations. (c) Impact of dropout rate.

(d) Impact of learning rate. (e) Impact of training dataset.

Figure 7: Evaluation results of cricket dataset.

the improvement of our method from 10% to 90% is remarkable. More specifically, when the proportion of the training dataset increases from 10% to 90%, the accuracy improvement of our method is 40.1% while the improvement of CNN, SRC, SVM and KNN are 34.7%, 29.3%, 26.7% and 22.4%, respectively. In this experiment, we do not test SVM-MFCC and KNN-MFCC because their accuracy is poor as will be shown later.

4.2.2. Comparison With Other Methods

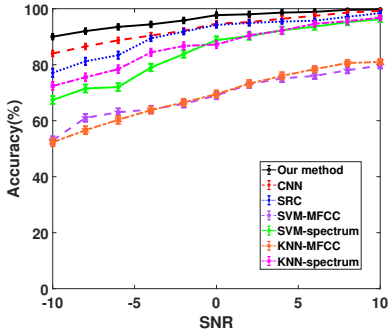
We now compare the performance of proposed scheme with previous approaches. As mentioned above, we compare the accuracy of the proposed system with conventional CNN, SRC, SVM-MFCC, SVM-spectrum, KNN-MFCC and KNN-spectrum. The MFCC of each window is calculated by transforming the power spectrum of each window into the logarithmic mel-frequency spectrum. We calculate the accuracy of different methods under different SNRs by adding different scales of environmental noise.

As we can see from Figure 5(f), SVM-MFCC and KNN-MFCC performs the worst which suggests that different frog species are not distinguishable in MFCC feature space. The results also explains why MFCC-based methods usually requires other carefully selected features [28]. We find that when the animal voice is overwhelmed by environmental noise (low SNR), the accuracy of our system is significantly higher than the other methods. For example, when $SNR = -6dB$, the accuracy of our method is 12% higher than CNN, 41% higher than SRC, 70% higher than SVM-MFCC, 53.9% higher than SVM-spectrum, 74.3% higher than KNN-MFCC, and 68% higher than KNN-spectrum. The robustness to noise makes the proposed system suitable for real deployment in noisy environments. Moreover, the results also indicate that our system needs less sensors to cover a certain area because our system can classify low SNR signals which are usually collected from longer distance.

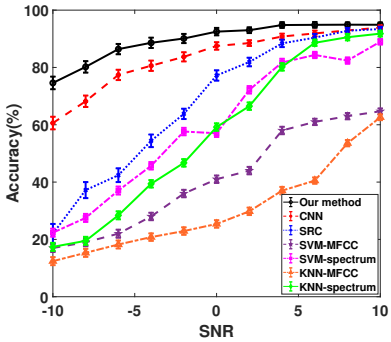
To take a closer look at the result, we summarize the results of different methods in Table 2 and plot confusion matrix in Figure 6 when SNR is -6dB. We can see that each class can achieve high accuracy and the overall average accuracy is 94.7%.

Table 3: Performance of different methods on cricket dataset (SNR=-6dB).

	Our method	CNN	SRC	SVM-MFCC	SVM-Spectrum	KNN-MFCC	KNN-Spectrum
Accuracy	**86.4%**	76.6%	42.4%	22.1%	36.8%	19.4%	28.5%
Precision	**86.9%**	76.2%	42.5%	23.6%	36.3%	18.2%	28.6%
Recall	**85.1%**	75.3%	41.2%	22.7%	37.3%	19.9%	29.8%
F1-score	**86.1%**	74.6%	41.7%	21.8%	38.1%	19.5%	29.5%



(a) Two-class classification.



(b) Twenty-class classification.

Figure 8: 2 class classification vs 20 class classification.

4.3. Performance of Cricket Dataset

Similar to Frog dataset, we also evaluate the impact of window size, the number of iterations, the dropout rate and learning rate in CNN, and the size of training dataset using Cricket dataset. The procedures are the same as above and the results are shown in Figure 8. We can see that it shows similar patterns as Frog dataset which suggests that the proposed framework is robust to different species. In terms of the dropout rate and learning rate, the optimal values for dropout rate and learning rate are 0.7 and 0.0005 which is slightly different from that of Frog dataset.

As mentioned in [1], the cricket dataset consists of twenty species of insects, eight of which are Gryllidae and twelve of which are Tettigoniidae. Thus, we can treat the problem as either a two-class genus level problem, or twenty-class species level problem. We first treat the classification as a two-class level problem and calcualte the accuracy of different methods under different SNRs. From the results in Figure 8(a), we can see that our method, SRC, SVM-spectrum and KNN-spectrum can achieve high accuracy. However, our method still outperforms all the other classifiers. Thereafter, we treat the classification as a twenty-class species level problem and plot the accuracy of different methods in Figure 8(a). We can see that the proposed method significantly outperforms the other methods when SNR is low. Table 3 summarizes the results of each method in detail. The results above demonstrate the advantage of our method in classifying more species in noisy environment.

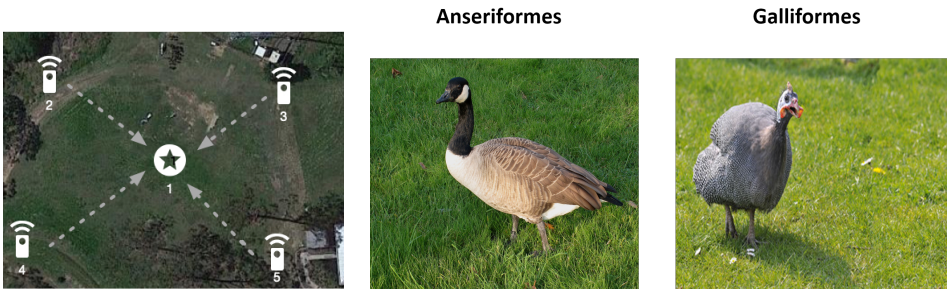
5. Case Study on Testbed

To validate the feasibility of the proposed framework in real environment, we implement the system on an outdoor ASN testbed which is located in Brisbane, Australia. As shown in Figure 9(a), the testbed is composed of five nodes which are configured as *Ad-hoc* mode with a star network topology. Its task is to evaluate the system's capability of recognizing bird vocalization in real world environment.

Table 4: Power Consumption.

Module	**Consumption (W)**
CPU	2.05
CPU + microphone	2.1
CPU + Wifi (idle)	2.45
CPU + Wifi (Rx)	2.67
CPU + Wifi (Tx)	2.78

The hardware platform used in the testbed is based on a Pandaboard ES with an 1.2Ghz OMAP



(a) Network topology (b) Bird species

Figure 9: Testbed.

4460, 1GB Ram and 4GB SD-card. Additionally, Pandaboard includes an 802.11 interface for wireless connection. Microphones are connected to the Pandaboard via USB port to record bird voice with 24Khz sampling rate. All the nodes are connected via the local Wi-Fi network. The data collected from Node 2, 3, 4 and 5 will be first transferred to Node 1. Then, all the data will be uploaded from Node 1 to the local server. The acoustic date from different nodes are classified separately in the system.

In the testbed, each node is powered by a rechargeable battery (12V, 7.2Ah), and an optional solar panel (5W, 12V). The power consumption of each module is given in Table 4. Compared to SolarStore testbed [52] which consumes 10W (low load) and 15W (high load) energy, our testbed is approximately 3.5 to 5.4 times more energy efficient. Without solar panel, a node in our ASN testbed will run continuously for more than 31 hours, which is significantly longer than the previous platforms such as ENSBox [53]. We find that if a solar panel is exposed to direct sunlight for 8 hours per day, the node can maintain a 50% duty cycle at 85% solar charge efficiency.

The nodes use Network Time Protocol (NTP) for time synchronization. We use one node as the NTP server, and the other nodes as the NTP clients. The NTP clients send request for time synchronization every 10 seconds. The accuracy of time synchronization is about 25 ms, which is good enough for our distributed real-time system because the length of each testing signal segment is 400ms.

During deployment, we found that the recorded voice is deteriorated by wind. To solve this problem, we take two measures. First, we install foam and fur windscreen around each microphone.

Table 5: Computation time of local processing.

	Silence Removal	FFT
Time (ms)	20.38 ± 2.04	15.33 ± 0.63

Second, we apply a Butterworth high pass filter with 200Hz cut-off frequency to filter out unwanted noise. This is because most of the wind audio energy lies in the frequency band below 200Hz, while most of the vocalization energy of the birds is in the frequency band higher than 200Hz.

After implementing the proposed framework on the testbed, we calculate the computation time on the node's side and classification accuracy on the server's side. On the node's side, we find that the node in our testbed can process all the captured acoustic data in real time. From Table 6, we can see the silence removal and FFT take 20.38 ms and 15.33 ms, respectively.

In this study, we choose two common bird species in the area of interest: Anseriformes and Galliformes (Figure 9(b)). Our goal is to classify the voice into three classes: Anseriformes, Galliformes and others. The testbed runs for 30 days and the data is labeled manually. Table 6 lists the results of different methods for classification in the server. We find that the proposed system achieve 90.3% classification accuracy which outperforms other methods significantly. The results in turn suggest that the proposed framework is robust to environmental noise and can achieve high classification accuracy in real-world WASN. We also notice that the results of the case study is slightly lower than the simulation results in Section 4. This is because the public dataset are collected in a controlled manner and the signals are well trimmed and

Table 6: Performance on testbed.

	Our system	CNN	SRC	SVM-Spectrum	KNN-Spectrum
Accuracy	**90.3%**	84.4%	72.3%	65.7%	68.8%
Precision	**91.2%**	82.1%	72.6%	66.4%	69.2%
Recall	**89.4%**	84.6%	70.9%	65.6%	67.1%
F1-score	**91.1%**	83.7%	71.8%	66.4%	70.5%

processed. However, the data we used in our case study are collected in a totally automatic manner.

6. Conclusion

In this paper, we design and implement a CNN-based acoustic classification system for WASN. To improve the accuracy in noisy environment, we propose a multi-view CNN framework which contains three convolution operation with three different filter length in parallel in order to extract the short-, middle-, and long-term information at the same time. Extensive evaluations on two real datasets show that the proposed system significantly outperforms previous methods. To demonstrate the performance of the proposed system in real world environment, we conduct a case study by implementing our system in a public testbed. The results show that our system works well and can achieve high accuracy in real deployments. In our future work, we will deploy the proposed framework in wider area and evaluate its performance in different environments.

Acknowledgement

The work described in this paper was fully supported by a grant from City University of Hong Kong (Project No.7200642)

References

[1] Y. Hao, B. Campana, E. Keogh, Monitoring and mining animal sounds in visual space, Journal of insect behavior 26 (4) (2013) 466–493.

[2] J. Luque, D. F. Larios, E. Personal, J. Barbancho, C. León, Evaluation of mpeg-7-based audio descriptors for animal voice recognition over wireless acoustic sensor networks, Sensors 16 (5) (2016) 717.

[3] I. F. Akyildiz, D. Pompili, T. Melodia, Underwater acoustic sensor networks: research challenges, Ad hoc networks 3 (3) (2005) 257–279.

[4] W. Hu, N. Bulusu, C. T. Chou, S. Jha, A. Taylor, V. N. Tran, Design and evaluation of a hybrid sensor network for cane toad monitoring, ACM Transactions on Sensor Networks (TOSN) 5 (1) (2009) 4.

[5] D. J. Nowak, J. E. Pasek, R. A. Sequeira, D. E. Crane, V. C. Mastro, Potential effect of anoplophora glabripennis (coleoptera: Cerambycidae) on urban trees in the united states, Journal of economic entomology 94 (1) (2001) 116–122.

[6] S. E. Anderson, A. S. Dave, D. Margoliash, Template-based automatic recognition of birdsong syllables from continuous recordings, The Journal of the Acoustical Society of America 100 (2) (1996) 1209–1219.

[7] J. A. Kogan, D. Margoliash, Automated recognition of bird song elements from continuous recordings using dynamic time warping and hidden markov models: A comparative study, The Journal of the Acoustical Society of America 103 (4) (1998) 2185–2196.

[8] S. Fagerlund, Bird species recognition using support vector machines, EURASIP Journal on Applied Signal Processing 2007 (1) (2007) 64–64.

[9] G. Guo, S. Z. Li, Content-based audio classification and retrieval by support vector machines, IEEE transactions on Neural Networks 14 (1) (2003) 209–215.

[10] C.-J. Huang, Y.-J. Yang, D.-X. Yang, Y.-J. Chen, Frog classification using machine learning techniques, Expert Systems with Applications 36 (2) (2009) 3737–3743.

[11] M. A. Acevedo, C. J. Corrada-Bravo, H. Corrada-Bravo, L. J. Villanueva-Rivera, T. M. Aide, Automated classification of bird and amphibian calls using machine learning: A comparison of methods, Ecological Informatics 4 (4) (2009) 206–214.

[12] R. Banerjee, M. Mobashir, S. D. Bit, Partial dct-based energy efficient compression algorithm for wireless multimedia sensor network, in: Proceedings of the 2014 IEEE International Conference on Electronics, Computing and Communication Technologies (IEEE CONECCT), IEEE, 2014, pp. 1–6.

[13] I. Dutta, R. Banerjee, S. D. Bit, Energy efficient audio compression scheme based on red black wavelet lifting for wireless multimedia sensor network, in: Proceedings of the 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI), IEEE, 2013, pp. 1070–1075.

[14] J. J. Diaz, E. F. Nakamura, H. C. Yehia, J. Salles, A. A. Loureiro, On the use of compressive sensing for the reconstruction of anuran sounds in a wireless sensor network, in: Proceedings of the 2012 IEEE International Conference on Green Computing and Communications (GreenCom), IEEE, 2012, pp. 394–399.

[15] B. Wei, M. Yang, Y. Shen, R. Rana, C. T. Chou, W. Hu, Real-time classification via sparse representation in acoustic sensor networks, in: Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems (Sensys), ACM, 2013, p. 21.

[16] Y. Shen, W. Hu, M. Yang, B. Wei, S. Lucey, C. T. Chou, Face recognition on smartphones via optimised sparse representation classification, in: Proceedings of

the 13th international symposium on Information processing in sensor networks, IEEE Press, 2014, pp. 237–248.

[17] W. Xu, Y. Shen, N. Bergmann, W. Hu, Sensor-assisted face recognition system on smart glass via multi-view sparse representation classification, in: Proceedings of the 2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN), IEEE, 2016, pp. 1–12.

[18] Y. Sun, Y. Chen, X. Wang, X. Tang, Deep learning face representation by joint identification-verification, in: Advances in neural information processing systems, 2014, pp. 1988–1996.

[19] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al., Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, IEEE Signal processing magazine 29 (6) (2012) 82–97.

[20] A. Graves, A.-r. Mohamed, G. Hinton, Speech recognition with deep recurrent neural networks, in: Proceedings of the 2013 ieee international conference on Acoustics, speech and signal processing (icassp), IEEE, 2013, pp. 6645–6649.

[21] R. Collobert, J. Weston, A unified architecture for natural language processing: Deep neural networks with multitask learning, in: Proceedings of the 25th international conference on Machine learning, ACM, 2008, pp. 160–167.

[22] Y. Wang, M. Huang, L. Zhao, et al., Attention-based lstm for aspect-level sentiment classification, in: Proceedings of the 2016 conference on empirical methods in natural language processing, 2016, pp. 606–615.

[23] G. Jahns, W. Kowalczyk, K. Walter, Sound analysis to recognize different animals, IFAC Proceedings Volumes 30 (26) (1997) 169–173.

[24] D. G. Preatoni, M. Nodari, R. Chirichella, G. Tosi, L. A. Wauters, A. Martinoli, Identifying bats from time-expanded recordings of search calls: comparing classification methods, The Journal of wildlife management 69 (4) (2005) 1601–1614.

[25] M. Hodon, P. Šarafín, P. Ševčík, Monitoring and recognition of bird population in protected bird territory, in: 2015 IEEE Symposium on Computers and Communication (ISCC), IEEE, 2015, pp. 198–203.

[26] P. A. Eliopoulos, I. Potamitis, D. C. Kontodimas, Estimation of population density of stored grain pests via bioacoustic detection, Crop Protection 85 (2016) 71–78.

[27] S. Ntalampiras, Automatic acoustic classification of insect species based on directed acyclic graphs, The Journal of the Acoustical Society of America 145 (6) (2019) EL541–EL546.

[28] G. Vaca-Castaño, D. Rodriguez, Using syllabic mel cepstrum features and k-nearest neighbors to identify anurans and birds species, in: 2010 IEEE workshop On Signal processing systems (SIPS), IEEE, 2010, pp. 466–471.

[29] Y. Sun, H. Qi, Dynamic target classification in wireless sensor networks, in: 2008 19th International Conference on Pattern Recognition, IEEE, 2008, pp. 1–4.

[30] Y. Wen, K. Zhang, Z. Li, Y. Qiao, A discriminative feature learning approach for deep face recognition, in: European Conference on Computer Vision, Springer, 2016, pp. 499–515.

[31] K. Chen, L. Yao, X. Wang, D. Zhang, T. Gu, Z. Yu, Z. Yang, Interpretable parallel recurrent neural networks with convolutional attentions for multi-modality activity modeling, Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN-18) (2018).

[32] C. Luo, X. Feng, J. Chen, J. Li, W. Xu, W. Li, L. Zhang, Z. Tari, A. Y. Zomaya, Brush like a dentist: Accurate monitoring of toothbrushing via wrist-worn gesture sensing, in: INFOCOM, IEEE, 2019, pp. 1234–1242.

[33] X. Zhang, L. Yao, C. Huang, S. Wang, M. Tan, G. Long, C. Wang, Multi-modality sensor data classification with selective attention, in: Proceedings of the 27th International Joint Conference on Artificial Intelligence IJCAI-18, International Joint Conferences on Artificial Intelligence Organization, 2018, pp. 3111–3117.

[34] G. Lan, W. Xu, D. Ma, S. Khalifa, M. Hassan, W. Hu, Entrans: Leveraging kinetic energy harvesting signal for transportation mode detection, IEEE Transactions on Intelligent Transportation Systems (2019).

[35] W. Xu, X. Feng, J. Wang, C. Luo, J. Li, Z. Ming, Energy harvesting-based smart transportation mode detection system via attention-based lstm, IEEE Access (2019).

[36] X. Zhang, L. Yao, S. S. Kanhere, Y. Liu, T. Gu, K. Chen, Mindid: Person identification from brain waves through attention-based recurrent neural network, Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 2 (3) (2018) 149.

[37] N. D. Lane, P. Georgiev, Can deep learning revolutionize mobile sensing?, in: Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications, ACM, 2015, pp. 117–122.

[38] N. D. Lane, S. Bhattacharya, P. Georgiev, C. Forlivesi, L. Jiao, L. Qendro, F. Kawsar, Deepx: A software accelerator for low-power deep learning inference on mobile devices, in: Proceedings of the 15th International Conference on Information Processing in Sensor Networks, IEEE Press, 2016, p. 23.

[39] Y.-J. Zhang, J.-F. Huang, N. Gong, Z.-H. Ling, Y. Hu, Automatic detection and classification of marmoset vocalizations using deep and recurrent neural networks, The Journal of the Acoustical Society of America 144 (1) (2018) 478–487.

[40] T. P. Oikarinen, K. Srinivasan, O. Meisner, J. B. Hyman, S. Parmar, R. Desimone, R. Landman, G. Feng, Deep convolutional network for animal sound classification and source attribution using dual audio recordings, bioRxiv (2018) 437004.

[41] S. Ntalampiras, Bird species identification via transfer learning from music genres, Ecological informatics 44 (2018) 76–81.

[42] I. Potamitis, Deep learning for detection of bird vocalisations, arXiv preprint arXiv:1609.08408 (2016).

[43] H. V. Koops, J. Van Balen, F. Wiering, A deep neural network approach to the lifeclef 2014 bird task, CLEF2014 Working Notes 1180 (2014) 634–642.

[44] H. Goëau, H. Glotin, W.-P. Vellinga, R. Planqué, A. Joly, Lifeclef bird identification task 2016: The arrival of deep learning, 2016.

[45] K. C. Barr, K. Asanović, Energy-aware lossless data compression, ACM Transactions on Computer Systems (TOCS) 24 (3) (2006) 250–291.

[46] W. Xu, J. Y. Kim, W. Huang, S. Kanhere, S. Jha, W. Hu, Measurement, characterization and modeling of lora technology in multi-floor buildings, IEEE Internet of Things Journal (2019).

[47] D. C. Ciresan, U. Meier, L. M. Gambardella, J. Schmidhuber, Convolutional neural network committees for handwritten character classification, in: Proceedings of the 2011 International Conference on Document Analysis and Recognition (ICDAR), IEEE, 2011, pp. 1135–1139.

[48] R. Ning, C. Wang, C. Xin, J. Li, H. Wu, Deepmag: Sniffing mobile apps in magnetic field through deep convolutional neural networks, in: Proceedings of the 2018 IEEE International Conference on Pervasive Computing and Communications (PerCom), IEEE, 2018, pp. 1–10.

[49] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: towards real-time object detection with region proposal networks, IEEE Transactions on Pattern Analysis & Machine Intelligence (6) (2017) 1137–1149.

[50] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1–9.

[51] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).

[52] Y. Yang, L. Wang, D. K. Noh, H. K. Le, T. F. Abdelzaher, Solarstore: enhancing data reliability in solar-powered storage-centric sensor networks, in: MobiSys, ACM, 2009, pp. 333–346.

[53] L. Girod, M. Lukac, V. Trifa, D. Estrin, The design and implementation of a self-calibrating distributed acoustic sensing platform, in: SenSys, ACM, 2006, pp. 71–84.